

CLAIMS

What is claimed is:

1. In computer system that is included in a topology for synchronizing a plurality of replicas, the computer system including a source replica from among the plurality of replicas, the physical layout of items stored at the source replica differing from the physical layout of corresponding items stored at one or more other replicas, a method for synchronizing replicas comprising:

an act of receiving a synchronization indication that the source replica is to be synchronized with the destination replica from among the plurality of replicas;

an act of determining if items in the physical layout at the source replica have changed;

an act of mapping any changed items in the physical layout at the source replica to a logical view, the logical view being substantially similar to logical views at one more other replicas in the topology, so as to increase the efficiency of synchronizing the source replica with the one or more other replicas, or to avoid any changes to the source replica when a new replica with different physical storage format is introduced into the sync topology; and

an act of sending at least one mapped item from the source replica to the destination replica.

2. The method of claim 1, further comprising:

an act of defining a source folder at the source replica, the source folder including a plurality of items, the source folder corresponding to a destination folder at the destination replica, the destination folder including a corresponding plurality of items; and

an act of sending at least a portion of the plurality of items.

3. The method of claim 1, the source replica comprising a data store layer and a synchronization layer, the data store layer comprising a local change tracker, the local change tracker maintaining local change enumerations associated with the items in the physical layout at the source replica and the synchronization layer comprising a synchronization change tracker, the synchronization change tracker maintaining versions and synchronization local change enumerations associated with the items mapped to the logical view of the source replica, the act of determining comprising an act of comparing the local change enumerations and synchronization local change enumerations.

4. The method of claim 3, wherein the act of determining if items in the physical layout at the source replica have changed comprising an act of determining that if the local change enumerations are different than the synchronization local change enumerations, items in the physical layout have changed.

5. The method of claim 3, wherein the act of determining if items in the physical layout at the source replica have changed comprising an act of deciding that if the local change enumerations are the same as the synchronization local change

enumerations, items associated with the local change enumerations and synchronization local change enumerations have not changed.

6. The method of claim 5, the versions comprising replica IDs and replica change enumerations, the replica IDs identifying replicas within the topology and the replica-change enumerations identifying chronological orders that items were changed at the plurality of replicas.

7. The method of claim 6, further comprising:

an act of identifying that an item mapped to the logical view comprises a change made by a replica identified by the replica ID of the version associated with the item mapped to the logical view.

8. The method of claim 6, the act of identifying comprising an act of identifying that an item mapped to the logical view is associated with a replica ID assigned by one of the plurality of replicas that assigns versions to items changed at a replica that made the change to the item mapped to the logical view.

9. The method of claim 3, the topology comprising a custom replica that comprises custom items and custom change enumerations associated with the custom items, the custom change enumerations being in a different format than the versions further comprising the step of mapping the logical view at the source replica to a custom view, the custom view correlating the versions and the custom change enumerations.

10. The method of claim 1, further comprising:

an act of defining a change unit in a logical schema, the change unit defining the granularity of an item, prior to the act of receiving the synchronization indication;

an act of defining a consistency unit in the logical schema, defining that the changes of one or more items of the defined granularity must be received in the same synchronization in order for any of the one or more items to be updated in a replica; and

an act of compiling the logical schema into at least a catalog, the catalog performing the act of mapping.

11. The method of claim 10, wherein the act of compiling generates procedures or functions code, the procedures or functions code directing the source replica where to store items in the physical layout.

12. The method of claim 1, further comprising:

an act of installing a catalog prior to receiving the synchronization indication, the catalog configured to map items from a physical layout to a logical view.

13. The method of claim 12, further comprising:

an act of installing procedures or functions code, the procedures or functions code directing the source replica where to store items in the physical layout.

WORKMAN NYDEGGER
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

14. In computer system that is included in a topology for synchronizing a plurality of replicas, the computer system including a source replica from among the plurality of replicas, the physical layout of the items stored at the source replica differing from the physical layout of corresponding items stored at one or more other replicas, a method for synchronizing replicas comprising:

a step for identifying items from a physical layout that are to be mapped to a logical view;

an act of mapping any changed items in the physical layout at the source replica to a logical view, the logical view being substantially similar to logical views at one more other replicas in the topology, so as to increase the efficiency of synchronizing the source replica with the one or more other replicas, or to avoid any changes to the source replica when a new replica with different physical storage format is introduced into the sync topology; and

an act of sending at least one mapped item from the source replica to the destination replica.

WORKMAN NYDEGGER
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

15. A computer system configured to synchronize data with other computer systems in a topology, the computer system comprising:

one or more processing units; and

one or more computer-readable media having stored thereon data, the data comprising:

a data store layer, the data store layer comprising:

a plurality of items; and

a local change tracker that maintains local change enumerations for the plurality of items; and

a synchronization layer, the synchronization layer comprising:

a logical view that is substantially similar to a logical views at one or more other computer systems in the topology, the logical view representing a mapping of one or more items from the plurality of items; and

a synchronization change tracker, the synchronization change tracker maintaining versions and synchronization local change enumerations for the one or more items mapped to the logical view.

16. The computer system of claim 15 further, the logical view comprises:

a change unit, the change unit defines the boundaries of an item; and

a consistency unit, the consistency unit comprises a plurality of change units defining a plurality of items the changes of which all must be received in the same synchronization by other computer systems in the topology from the

computer system in order for any of the plurality of items to be updated in the other computer systems.

17. The computer system of claim 15, the data store layer comprises a folder enumerating items in the data store, the items enumerated by the folder are applied into a corresponding folder at another computer system in the topology.

18. The computer system of claim 15, wherein the plurality of items in the data store layer are stored in a table.

19. The computer system of claim 15, wherein the plurality of items in the data store layer are stored in a plurality of tables.

20. The computer system of claim 15, wherein the plurality of items in the data store layer are stored in files.

21. The computer system of claim 15, wherein the synchronization layer is adapted to determine if an item in the data store was made at the computer system and not synchronized; or by some other computer system or previously synchronized by comparing the local change enumeration and the synchronization local change enumeration.

22. The computer system of claim 21, wherein the version comprises a replica ID and a change enumeration, the replica ID corresponds to a computer system

in the topology and the change enumeration corresponds to a chronological time when the change to the item was made.

23. The computer system of claim 22, wherein the synchronization layer can determine what computer system made the change to the item by examining the replica ID.

24. The computer system of claim 15, wherein the synchronization layer comprises a custom view, the custom view is mapped from the logical view and, the custom view correlates custom change enumerations existing at another computer system with local change enumerations at the data store.

WORKMAN NYDEGGER
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

25. At a compiling computer system, a method for generating a mapping of the physical layout of items in a data store to a logical view, the method comprising:
- an act of accessing core code that can be used to facilitate compilation of logical schemas;
 - an act of accessing a logical schema, the logical schema including:
 - a change unit that defines the granularity of an item; and
 - a consistency unit, defining the changes of one or more items of the defined granularity that must be received in the same synchronization in order for any of the one or more items to be updated in a replica; and
 - an act of utilizing the core code to compile the logical schema into at least a catalog, the catalog mapping items from a physical layout to a logical view, the logical view being substantially similar to a logical view at one or more other computer systems.
26. The method of claim 25, further comprising:
- an act of configuring the compiling computer system to send items to at least one of the one or more other computer systems.
27. The method of claim 25, further comprising:
- an act of installing the catalog to a computer system for use with a data store configured in the physical layout.
28. The method of claim 25, further comprising:

an act of sending the catalog to a computer system for use with a replica that is configured to synchronize data according to the logical view.

29. The method of claim 27, further comprising:

an act of dividing the computer system into a data store layer and a synchronization layer.

30. The method of claim 29, wherein the act of compiling further creates procedures or functions code, the procedures or functions code is adapted to arrange items in the data store layer and further comprising an act of locating the procedures or functions code at the data store layer.

31. The method of claim 29, further comprising:

an act of locating the catalog in the synchronization layer.

32. The method of claim 29, further comprising:

an act of storing a local change tracker at the data store layer, the local change tracker maintaining local change enumerations for items stored in the data store layer;

an act of storing a synchronization change tracker at the synchronization layer, the synchronization change tracker maintaining versions and synchronization local change enumerations for the items stored in the synchronization layer; and

wherein by comparing the local change tracker with the synchronization local change tracker, the computer system can determine if an item stored in the data store layer of the computer system should be sent and thus mapped to the logical view in synchronization.

33. The method of claim 32, wherein if the local change enumeration and the synchronization local change enumeration comprise different values, then the item stored in the data store layer of the computer system should be mapped to the logical view.

34. The method of claim 32, wherein if the local change enumeration and the synchronization local change enumeration comprise the same value, then the item stored in the data store layer of the computer system does not need to be mapped to the logical view.

35. The method of claim 32, wherein the versions comprising replica IDs correspond to a computer systems in the topology and change enumerations corresponding to a chronological order that a change was made.

36. The method of claim 35, wherein the replica ID corresponds to a computer system that changed the item.

37. The method of claim 35, wherein the replica ID corresponds to a computer system that assigns versions for changes made at computer systems other than the computer system that assigns versions for changes.

38. The method of claim 29, further comprising:

an act of storing a folder which is exposed to a user interface as a generic way to organize data items, in the data store layer, the folder being adapted to group items together.

39. The method of claim 38, wherein the replica is configured to send items grouped in the folder.

WORKMAN NYDEGGER
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

40. A computer program product for use in computer system that is included in a topology for synchronizing a plurality of replicas, the computer system including a source replica from among the plurality of replicas, the physical layout of items stored at the source replica differing from the physical layout of corresponding items stored at one or more other replicas, the computer program product for implementing a method for synchronizing replicas, the computer program product comprising one or more computer-readable media having stored thereon computer executable instructions that, when executed by a processor, cause the computer system to perform the following:

receive a synchronization indication that the source replica is to be
synchronized with the destination replica from among the plurality of replicas;

determine if items in the physical layout at the source replica have
changed;

map any changed items in the physical layout at the source replica to a
logical view, the logical view being substantially similar to logical views at one
more other replicas in the topology, so as to increase the efficiency of
synchronizing the source replica with the one or more other replicas; and

send at least one mapped item from the source replica to the destination
replica.

41. A computer program product for use in a compiling computer system, the computer program product for implementing a method for generating a mapping of the physical layout of items in a data store to a logical view, the computer program product comprising one or more computer-readable media having stored thereon computer executable instructions that, when executed by a processor, cause the compiling computer system to perform the following:

access core code that can be used to facilitate compilation of logical schemas;

access a logical schema, the logical schema including:

a change unit that defines the granularity of an item; and

a consistency unit, defining the changes of one or more items of the defined granularity must be received in the same synchronization in order for any of the one or more items to be updated in a replica; and

utilize the core code to compile the logical schema into at least a catalog, the catalog mapping items from a physical layout to a logical view, the logical view being substantially similar to a logical view at one or more other computer systems.